Федеральное государственное бюджетное образовательное учреждение высшего образования

«Сибирский государственный университет телекоммуникаций и информатики»

(СибГУТИ)

09.03.01 "Информатика и вычислительная техника"

профиль "Программное обеспечение средств вычислительной техники и автоматизированных систем"

**ОТЧЕТ**

по учебной практике

на кафедре Прикладной Математики и Кибернетики

Выполнил:

студент гр. ИП-815 \_\_\_\_\_\_\_\_\_\_\_\_\_/Захаров М.Д./

ФИО студента

«04» июля 2020г.

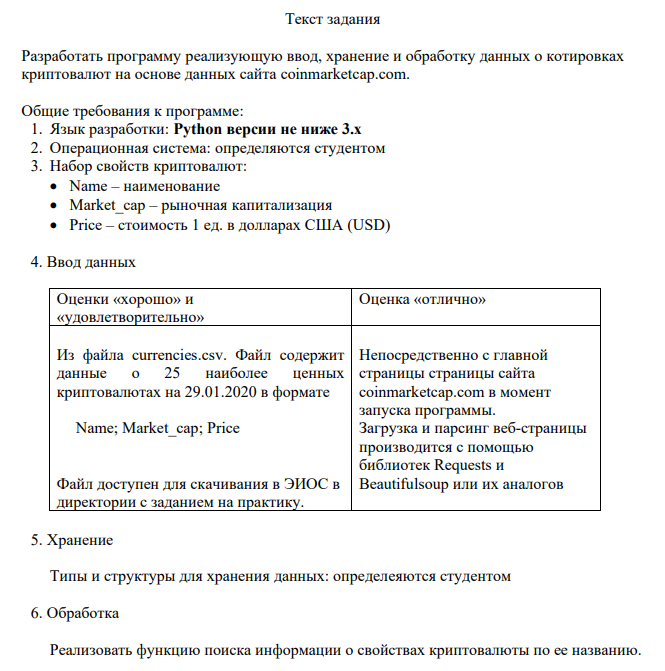
Руководитель практики

доцент каф. ПМиК \_\_\_\_\_\_\_\_\_\_\_\_\_/Приставка П.А./

«04» июля2020г. Оценка\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Новосибирск 2020г.

**Условие задачи**

****

**Описание алгоритма**

Алгоритм программы устроен так, что есть ряд функций, которые выполняют свои задачи, и главная функция def parse() , в которой вызовы этих второстепенных функций и обработка возвращаемых ими значений.

def get\_content(html)-функция, в которой происходит поиск нужных нам значений на сайте и они собираются в список словарей. На вход подается html код страниц.

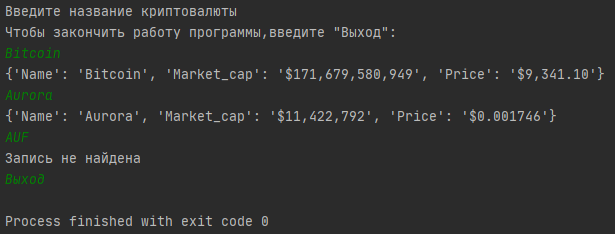
def search(Name, valutes)- эта функция по введенным данным ищет совпадение введенного пользователем названия криптовалюты со значениями в списке словарей и возвращает словарь где записаны все данные для искомой валюты, иначе возвращает None.Принимает на вход название криптовалюты и список словарей.

def parse()-главная функция в программе, которая сначала собирает html код со всех страниц сайта, затем вызывает второстепенные функции и реализовывает интерфейс программы (пользовательский ввод/вывод, обработка неверно введенных значений и выход из программы).

**Листинг программы**

import requests  
from bs4 import BeautifulSoup  
  
  
  
URL = 'https://coinmarketcap.com/'  
valutes = []  
  
  
def get\_content(html):  
 soup = BeautifulSoup(html, 'html.parser')  
 itemsN = soup.find\_all('tr', class\_='cmc-table-row')  
 itemsM = soup.find\_all('td', class\_='cmc-table\_\_cell cmc-table\_\_cell--sortable cmc-table\_\_cell--right cmc-table\_\_cell--sort-by\_\_market-cap')  
 itemsP = soup.find\_all('td', class\_='cmc-table\_\_cell cmc-table\_\_cell--sortable cmc-table\_\_cell--right cmc-table\_\_cell--sort-by\_\_price')  
  
 for itemN,itemM,itemP in zip(itemsN, itemsM, itemsP):  
 valutes.append({  
 'Name': itemN.find('a', class\_='cmc-link').get\_text(strip=True),  
 'Market\_cap': itemM.find('div').get\_text(strip=True),  
 'Price': itemP.find('a', class\_='cmc-link').get\_text(strip=True)  
 })  
  
 return(valutes)  
  
  
def search(Name, valutes):  
  
 return next((item for item in valutes if item.get("Name") and item["Name"] == Name),None)  
  
def parse():  
 max\_pages = 26  
 pages = []  
 for x in range(1, max\_pages + 1):  
 pages.append(requests.get(URL + str(x)))  
  
 for r in pages:  
 html = r  
  
  
 get\_content(html.text)  
  
 print("Введите название криптовалюты")  
 print('Чтобы закончить работу программы,введите "Выход":')  
  
 while(1):  
  
 s = input()  
 if(s!='Выход'):  
 if (search(s, valutes))!=None:  
 print(search(s, valutes))  
 else:  
 print('Запись не найдена')  
 else:  
 break  
  
  
parse()

**Результаты тестирования**

****

**Список использованных источников**

<https://www.8host.com/blog/rabota-s-veb-dannymi-s-pomoshhyu-requests-i-beautiful-soup-v-python-3/>

<https://www.python.org/doc/>

<https://requests.readthedocs.io/en/master/>

<https://www.crummy.com/software/BeautifulSoup/>